

SYSTEMS FOR ANALYZING MICROTISSUE ARRAYS

Background of the Invention

Tissue microarrays are a method of harvesting small disks of tissue from a range of standard histologic sections and arranging them on a recipient paraffin block such that hundreds or thousands of disks can be analyzed simultaneously. This technique allows maximization of tissue resources by analysis of small core biopsies of blocks, rather than complete sections. A carefully planned array of tissues can be constructed with cases from pathology tissue block archives, such that a 20-year survival analysis can be performed on a cohort of 600 or more patients by use of only a few microliters of antibody.

Tissue microarray technology has numerous advantages in addition to tissue amplification. For example, each specimen is treated in an identical manner. Like conventional formalin-fixed paraffin embedded material, tissue microarrays are amenable to a wide variety of techniques, including histochemical stains, immunologic stains with either chromogenic or fluorescent visualization, *in situ* hybridization (including messenger RNA *in situ* hybridization and fluorescence *in situ* hybridization) and even microdissection techniques. For each of these protocols conventional sections can have substantial slide-to-slide variability associated with processing 300 slides (e.g. 20 batch of 15 slides). By contrast, the tissue microarrays allow an entire cohort to be analyzed on a single slide. Thus, reagent concentrations are identical for each case, as are incubation times and temperatures and wash conditions. Antigen retrieval can be another significant variable in conventional sections, which is mitigated by the identical treatment of specimens in a microarray. As a further advantage, only a few microliters of reagent may be required to analyze an entire

cohort in a microarray. This advantage raises the possibility of using tissue microarrays in certain screening procedures, such as hybridoma screening, where the protocol is not amenable to the use of conventional sections.

5 Currently, the primary method used to evaluate microarrays involves manual review of hundreds of tissue microarray ("TMA") cores under a microscope, while subjectively evaluating and scoring the signal at each location. An alternate, but less utilized approach is to sequentially digitize specimens for subsequent assessment. Both procedures involve manually and systematically reviewing the TMA sample under the microscope, which is a
10 slow, tedious process, and which is especially error-prone because it is easy to loose track of a current array while navigating among the regularly arranged specimens. This is especially true at higher (e.g. 20x) magnifications.

 Tissue microarrays also present some special problems such as heterogeneity of
15 tissue sections, sub-cellular localization of staining, and background signal. Depending on the type of tumor or tissue section analyzed, the area of interest may represent nearly the entire disk or only a small percentage thereof. For example, a pancreatic carcinoma or lobular carcinoma of the breast with substantial desmoplastic response may show stromal tissue representing a large percentage of the total area of the disk. If the goal of the assay is
20 to determine epithelial cell expression of a given marker, a protocol must be used that evaluates only that region of the disk. The protocol must not only be able to select the region of interest but also normalize it so that the expression level read from any given disk can be compared with that of other disks. Sub-cellular localization presents a host of additional

challenges when comparing nuclear or membranous stainings which are quite different from those in total cytoplasmic staining.

There remains a need for a systematic approach to collecting, analyzing, and storing
5 data from tissue microarrays.

Summary Of The Invention

The systems described herein autonomously image, analyze, and store data for samples in a tissue microarray. The system may include a tissue microarray, a robotic
10 microscope, and an imaging workstation that executes software to automatically control operation of the microscope to capture images from the microarray and analyze image results. A low magnification may be used to register samples within the microarray and obtain coordinates for each tissue specimen. Progressively higher magnifications may be used to analyze images of each registered specimen. Images and quantitative data from the
15 images may then be stored in a relational database for subsequent review. The system may be local, or may be Web-based for distributed control and sharing of results.

Brief Description Of Drawings

The foregoing and other objects and advantages of the invention will be appreciated
20 more fully from the following further description thereof, with reference to the accompanying drawings, wherein:

Fig. 1 shows a schematic diagram of the entities involved in an embodiment of a method and system disclosed herein;

Fig. 2 shows a block diagram of a server that may be used with the systems described herein;

Fig. 3 shows a page that may be used as a user interface; and

Fig. 4 is a flow chart of a process for capturing, processing, and storing images of disks in a tissue microarray.

Detailed Description of Certain Embodiments of the Invention

To provide an overall understanding of the invention, certain illustrative embodiments will now be described, including a system for automated analysis of a tissue microarray. However, it will be understood that the methods and systems described herein can be suitably adapted to any environment where a number of approximately regularly spaced specimens are to be visually inspected in some systematic fashion. For example, the systems and methods are applicable to a wide range of biological specimen images, and in particular to analysis or diagnosis involving cellular, or other microscopic, visual data. These and other applications of the systems described herein are intended to fall within the scope of the invention.

Figure 1 shows a schematic diagram of the entities involved in an embodiment of a method and system disclosed herein. In a system 100, one or more imaging devices 101, a plurality of clients 102, servers 104, and providers 108 are connected via an internetwork 110. It should be understood that any number of clients 102, servers 104, and providers 108 could participate in such a system 100. The system may further include one or more local area networks ("LAN") 112 interconnecting clients 102 through a hub 114 (in, for example,

a peer network such as Ethernet) or a local area network server 114 (in, for example, a client-server network). The LAN 112 may be connected to the internetwork 110 through a gateway 116, which provides security to the LAN 112 and ensures operating compatibility between the LAN 112 and the internetwork 110. Any data network may be used as the internetwork
5 110 and the LAN 112.

In one embodiment, the internetwork 110 is the Internet, and the World Wide Web provides a system for interconnecting imaging devices 101, clients 102 and servers 104 through the Internet 110. The internetwork 110 may include a cable network, a wireless network, and any other networks for interconnecting clients, servers and other devices.

10 As depicted, one of the imaging devices 101 may be connected to one of the clients 102, one of the servers 104, the hub 114 of the LAN 112, or directly to one of the providers 108, and may include suitable hardware and software for connecting to the internetwork 110 through any of the above devices or systems. One of the imaging devices 101 that may be used in the systems herein is a high-resolution color video camera, such as an Olympus
15 OLY-750 coupled to a Coreco, Oculus data acquisition board. This imaging device 101 may be used to gather images for the image database, as described in more detail below. Another one of the imaging devices 101 may be a robotic microscope, such as an Olympus AX70, allowing electronic control over a specimen stage, a light level, an objective lens, and a focus, as well as parameters of digitization such as rate and resolution. The imaging
20 devices 101 may be steered to an x-position and a y-position of a specimen through electronic control. One of the imaging devices 101 may be used to obtain a query image.

More generally, the term 'imaging device' as used herein should be understood to include cameras, microscopes, or any other device for capturing and/or providing an image in electronic form, and should further be understood to include to include a mass storage device or other device for providing a previously captured electronic image.

5 In the systems described herein, the imaging devices 101 are used to obtain images of tissue microarrays. A tissue microarray may be a block of paraffin or similar material having holes placed therein to receive tissue samples. The samples placed in the tissue microarray are typically placed in some regular pattern, such as a rectangular matrix of cores, possibly with rows and/or columns skipped at regular intervals to facilitate visual navigation
10 of the array. In such an embodiment, each core has an x-coordinate and a y-coordinate at or near the center of the core, which may be identified and used to locate the core as described below. Other regular or irregular patterns may also, or instead be used, provided each core can be located and revisited within the array. It will be appreciated that, while disks are a common geometry used for samples in a tissue microarray, other geometries are possible,
15 including regular and irregular geometric profiles, and may be used with the system described herein, provided they are amenable to punching of matching shapes in a tissue source (for taking samples) and the receiving material (e.g., paraffin). The terms 'disk' or 'core', as used herein, are intended to include any such geometry. The terms 'specimen' or 'biological specimen' are intended to refer to any biological (or inert control) material that
20 may be sampled and inserted into a tissue microarray.

An exemplary client 102 includes the conventional components of a client system,

such as a processor, a memory (e.g. RAM), a bus which couples the processor and the memory, a mass storage device (e.g. a magnetic hard disk or an optical storage disk) coupled to the processor and the memory through an I/O controller, and a network interface coupled to the processor and the memory, such as modem, digital subscriber line (“DSL”) card, cable
5 modem, network interface card, wireless network card, or other interface device capable of wired, fiber optic, or wireless data communications. One example of such a client 102 is a personal computer equipped with an operating system such as Microsoft Windows 2000, Microsoft Windows NT, Unix, Linux, and Linux variants, along with software support for Internet communication protocols. The personal computer may also include a browser
10 program, such as Microsoft Internet Explorer or Netscape Navigator, to provide a user interface for access to the Internet 110. Although the personal computer is a typical client 102, the client 102 may also be a workstation, mobile computer, Web phone, television set-top box, interactive kiosk, personal digital assistant, or other device capable of communicating over the Internet 110. As used herein, the term “client” is intended to refer
15 to any of the above-described clients 102, as well as proprietary network clients designed specifically for the systems described herein, and the term “browser” is intended to refer to any of the above browser programs or other software or firmware providing a user interface for navigating the Internet 110 and/or communicating with the medical image processing systems.

20 An exemplary server 104 includes a processor, a memory (e.g. RAM), a bus which couples the processor and the memory, a mass storage device (e.g. a magnetic or optical disk) coupled to the processor and the memory through an I/O controller, and a network interface

coupled to the processor and the memory. Servers may be organized as layers of clusters in order to handle more client traffic, and may include separate servers for different functions such as a database server, a file server, an application server, and a Web presentation server.

Such servers may further include one or more mass storage devices such as a disk farm or
5 a redundant array of independent disk ("RAID") system for additional storage and data integrity. Read-only devices, such as compact disc drives and digital versatile disc drives, may also be connected to the servers. Suitable servers and mass storage devices are manufactured by, for example, Compaq, IBM, and Sun Microsystems. As used herein, the term "server" is intended to refer to any of the above-described servers 104.

10 Focusing now on the internetwork 110, one embodiment is the Internet. The structure of the Internet 110 is well known to those of ordinary skill in the art and includes a network backbone with networks branching from the backbone. These branches, in turn, have networks branching from them, and so on. The backbone and branches are connected by routers, bridges, switches, and other switching elements that operate to direct data through
15 the internetwork 110. However, one may practice the present invention on a wide variety of communication networks. For example, the internetwork 110 can include interactive television networks, telephone networks, wireless data transmission systems, two-way cable systems, customized computer networks, interactive kiosk networks, or ad hoc packet relay networks.

20 One embodiment of the internetwork 110 includes Internet service providers 108 offering dial-in service, such as Microsoft Network, America OnLine, Prodigy and

CompuServe. It will be appreciated that the Internet service providers 108 may also include any computer system which can provide Internet access to a client 102. Of course, the Internet service providers 108 are optional, and in some cases, the clients 102 may have direct access to the Internet 110 through a dedicated DSL service, ISDN leased lines, T1
5 lines, digital satellite service, cable modem service, or any other high-speed connection to a network point-of-presence. Any of these high-speed services may also be offered through one of the Internet service providers 108.

In its present deployment as the Internet, the internetwork 110 consists of a worldwide computer network that communicates using protocols such as the well-defined
10 Transmission Control Protocol ("TCP") and Internet Protocol ("IP") to provide transport and network services. Computer systems that are directly connected to the Internet 110 each have a unique IP address. The IP address consists of four one-byte numbers (although a planned expansion to sixteen bytes is underway with IPv6). The four bytes of the IP address are commonly written out separated by periods such as "xxx.xxx.xxx.xxx". To simplify
15 Internet addressing, the Domain Name System ("DNS") was created. The DNS allows users to access Internet resources with a simpler alphanumeric naming system. A DNS name consists of a series of alphanumeric names separated by periods. For example, the name "www.umdj.edu" corresponds to a particular IP address. When a domain name is used, the computer accesses a DNS server to obtain the explicit four-byte IP address. It will be
20 appreciated that other internetworks 110 may be used with the invention. For example, the internetwork 110 may be a wide-area network, a local-area network, or corporate-area network.

To further define the resources on the Internet 110, the Uniform Resource Locator system was created. A Uniform Resource Locator (“URL”) is a descriptor that specifically defines a type of Internet resource along with its location. URLs have the following format:

resource-type://domain.address/path-name

5 where *resource-type* defines the type of Internet resource. Web documents are identified by the resource type “http” which indicates that the hypertext transfer protocol should be used to access the document. Other common resource types include “ftp” (file transmission protocol), “mailto” (send electronic mail), “file” (local file), and “telnet.” The *domain.address* defines the domain name address of the computer that the resource is located
10 on. Finally, the *path-name* defines a directory path within the file system of the server that identifies the resource. As used herein, the term “IP address” is intended to refer to the four-byte Internet Protocol address (or the sixteen-byte IPv6 address), and the term “Web address” is intended to refer to a domain name address, along with any resource identifier and path name appropriate to identify a particular Web resource. The term “address,” when used
15 alone, is intended to refer to either a Web address or an IP address.

In an exemplary embodiment, a browser, executing on one of the clients 102, retrieves a Web document at an address from one of the servers 104 via the internetwork 110, and displays the Web document on a viewing device, e.g., a screen. A user can retrieve and view the Web document by entering, or selecting a link to, a URL in the browser. The
20 browser then sends an http request to the server 104 that has the Web document associated

with the URL. The server 104 responds to the http request by sending the requested Web document to the client 102. The Web document is an http object that includes plain text, or ASCII, conforming to the HyperText Markup Language (“HTML”). Other markup languages are known and may be used on appropriately enabled browsers and servers, including the Dynamic HyperText Markup Language (“DHTML”), the Extensible Markup Language (“XML”), the Extensible Hypertext Markup Language (“XHML”), and the Standard Generalized Markup Language (“SGML”).

Each Web document may contain hyperlinks to other Web documents. The browser displays the Web document on the screen for the user and the hyperlinks to other Web documents are emphasized in some fashion such that the user can identify and select each hyperlink. To enhance functionality, a server 104 may execute programs associated with Web documents using programming or scripting languages, such as Perl, C, C++, or Java. A server 104 may also use server-side scripting languages such as ColdFusion from Allaire, Inc., or PHP. These programs and languages perform “back-end” functions such as transaction processing, database management, content searching, and implementation of application logic for applications. A Web document may also include references to small client-side applications, or applets, that are transferred from the server 104 to the client 102 along with a Web document and executed locally by the client 102. Java is one popular example of a programming language used for applets. The text within a Web document may further include (non-displayed) scripts that are executable by an appropriately enabled browser, using a scripting language such as JavaScript or Visual Basic Script. Browsers may further be enhanced with a variety of helper applications to interpret various media including

still image formats such as JPEG and GIF, document formats such as PS and PDF, motion picture formats such as AVI and MPEG, and sound formats such as MP3 and MIDI. These media formats, along with a growing variety of proprietary media formats, may be used to enrich a user's interactive and audio-visual experience as each Web document is presented through the browser. The term "page" as used herein is intended to refer to the Web document described above, as well as any of the above-described functional or multimedia content associated with the Web document.

Figure 2 shows a block diagram of a server that may be used with the systems described herein. In this embodiment, the server 104 includes a presentation server 200, an application server 202, and a database server 204. The application server 202 is connected to the presentation server 200. The database server 204 is also connected to the presentation server 200 and the application server 202, and is further connected to a database 206 embodied on a mass storage device. The presentation server 200 includes a connection to the internetwork 110. It will be appreciated that each of the servers may comprise more than one physical server, as required for capacity and redundancy, and it will be further appreciated that in some embodiments more than one of the above servers may be logical servers residing on the same physical device. One or more of the servers may be at a remote location, and may communicate with the presentation server 200 through a local area or wide area network. The term "host," as used herein, is intended to refer to any combination of servers described above that include a presentation server 200 for providing access to pages by the clients 102. The term "site," as used herein, is intended to refer to a collection of pages sharing a common domain name address, or dynamically generated by a common host,

or accessible through a common host (i.e., a particular page may be maintained on or generated by a second, remote or local server, but nonetheless be within a 'site').

2050220" 0242200T

The presentation server 200 provides an interface for one or more connections to the
5 internetwork 110, thus permitting more than one of the clients 102 (Fig. 1) to access the site
at the same time. In one embodiment, the presentation server 200 comprises a plurality of
enterprise servers, such as the ProLiant Cluster available from Compaq Computer Corp., or
a cluster of E250's from Sun Microsystems running Solaris 2.7. Other suitable servers are
known in the art and are and may be adapted to use with the systems described herein, such
10 as, for example, an iPlanet Enterprise Server 4.0 from the Sun/Netscape Alliance. The
presentation server 200 may also use, for example, Microsoft's .NET technology, or use a
Microsoft Windows operating system, with a "front end" written in Microsoft Active Server
Page ("ASP"), or some other programming language or server software capable of
integrating ActiveX controls, forms, Visual Basic Scripts, JavaScript, Macromedia Flash
15 Technology multimedia, e-mail, and other functional and multimedia aspects of a page.
Typically, the front end includes all text, graphics, and interactive objects within a page,
along with templates used for dynamic page creation. The presentation server 200 maintains
one or more connections to the Internet 110. Where there is substantial network traffic, the
connections are preferably provided by a tier one provider, i.e., one of the dozen or so
20 national/international Internet backbones with cross-national links of T3 speeds or higher,
such as MCI, UUNet, BBN Planet, and Digex.

A client 102 (Fig. 1) accessing an address hosted by the presentation server 200 will

receive a page from the presentation server 200 containing text, forms, scripts, active objects, hyperlinks, etc., which may be collectively viewed using a browser. Each page may consist of static content, i.e., an HTML text file and associated objects (*.avi, *.jpg, *.gif, etc.) stored on the presentation server, and may include active content including applets, scripts, and objects such as check boxes, drop-down lists, and the like. A page may be dynamically created in response to a particular client 102 request, including appropriate queries to the database server 204 for particular types of data to be included in a responsive page. It will be appreciated that accessing a Web page is more complex in practice, and includes, for example, a DNS request from the client 102 to a DNS server, receipt of an IP address by the client 102, formation of a TCP connection with a port at the indicated IP address, transmission of a GET command to the presentation server 200, dynamic page generation (if required), transmission of an HTML object, fetching additional objects referenced by the HTML object, and so forth.

The application server 202 provides the “back-end” functionality of the Web site, and includes connections to the presentation server 200 and the database server 204. In one embodiment, the presentation server 200 comprises an enterprise server, such as one available from Compaq Computer Corp., running the Microsoft Windows NT operating system, or a cluster of E250’s from Sun Microsystems running Solaris 2.7. The back-end software may be implemented using pre-configured e-commerce software, such as that available from Pandesic, to provide back-end functionality including transaction processing, billing, data management, financial transactions, order fulfillment, and the like. The application server 202 may include a software interface to the database server 204, as well

as a software interface to the front end provided by the presentation server 200. The application server 200 may also use a Sun/Netscape Alliance Server 4.0.

205020" 042200T

The database server 204 may be an enterprise server, such as one available from Compaq Computer Corp., running the Microsoft Windows NT operating system or a cluster of E250's from Sun Microsystems running Solaris 2.7, along with software components for database management. Suitable databases are provided by, for example, Oracle, Sybase, and Informix. The database server 204 may also include one or more databases 206, typically embodied in a mass-storage device. The databases 206 may include, for example, user interfaces, search results, search query structures, lexicons, user information, and the templates used by the presentation server to dynamically generate pages. It will be appreciated that the databases 206 may also include structured or unstructured data, as well as storage space, for use by the presentation server 200 and the application server 202. In operation, the database management software running on the database server 204 receives properly formatted requests from the presentation server 200, or the application server 202. In response, the database management software reads data from, or writes data to, the databases 206, and generates responsive messages to the requesting server. The database server 204 may also include a File Transfer Protocol ("FTP") or a Secure Shell ("SSH") server for providing downloadable files.

While the three tier architecture described above is one conventional architecture that may be used with the systems described herein, it will be appreciated that other architectures for providing data and processing through a network are known and may be used in addition

to, or in conjunction with, or in place of the described architecture. Any such system may be used, provided that it can support aspects of the image processing system described herein.

Figure 3 shows a page that may be used as a user interface. The page 300 may
5 include a header 302, a sidebar 304, a footer 306 and a main section 308, all of which may be displayed at a client 102 using a browser. The header 302 may include, for example, one or more banner advertisements and a title of the page. The sidebar 304 may include a menu of choices for a user at the client 102. The footer 306 may include another banner advertisement, and/or information concerning the site such as a “help” or “webmaster”
10 contact, copyright information, disclaimers, a privacy statement, etc. The main section 308 may include content for viewing by the user. The main section 308 may also include, for example, tools for electronically mailing the page to an electronic mail (“e-mail”) account, searching content at the site, and so forth. It will be appreciated that the description above is generic, and may be varied according to where a client 102 is within a Web site related to
15 the page, as well as according to any available information about the client 102 (such as display size, media capabilities, etc.) or the user.

A Web site including the page 300 may use cookies to track users and user information. In particular, a client 102 accessing the site may be accessed to detect whether the client 102 has previously accessed the page or the site. If the client 102 has accessed the
20 site, then some predetermined content may be presented to the client 102. If the client 102 does not include a cookie indicating that the client 102 has visited the site, then the client 102

may be directed to a registration page where information may be gathered to create a user profile. The client 102 may also be presented with a login page, so that a pre-existing user on a new client 102 may nonetheless bypass the registration page.

1 The site may provide other functionality to the client 102. For example, the site may
5 provide a search tool by which the client 102 may search for content within the site, or
content external to the site but accessible through the internetwork 110. As another example,
the site may display local or remote news items and stories that are topical to the site. The
site may provide an interface for structured queries to, browsing of, and review of images and
data in, the database that stores archived tissue microarrays. Tools may also be provided for
10 other network functions associated with the system, such as remotely initiating data capture
for a tissue microarray, manual control of a robotic microscope or other imaging device used
to obtain tissue microarray images, or manual control of an imaging device.

20 The interface may be embodied in any software and/or hardware client operating on
a client device, including a browser along with any suitable plug-ins, a Java applet, a Java
15 application, a C or C++ application, or any other application or group of applications
operating on a client device. In one embodiment, the user interface may be deployed through
a Web browser. In one embodiment, the user interface may be deployed as an application
running on a client device, with suitable software and/or hardware for access to an
internetwork. In these and other embodiments, certain image processing functions, as well
20 as database storage and management functions, may be distributed in any suitable manner
between a client device, one or more imaging devices, and one or more servers.

It will be appreciated that a number of enhancements may be provided to the user interface. For example, voice-activated commands may be provided. Voice communication between the user and computer may enable a user to navigate among digital archives of tissue microarrays or to direct the inspection of disk specimens, or “cores”, while they are viewed
5 with the robotic microscope. Valid voice commands may include, for example, “next core”, “current core”, “previous core”, and “where am I?”. The user can also direct the scope to move to a specific core location by indicating its row and column. For quality control purposes the system may support programmed screening of samples, in which each core in an array is retrieved and displayed to the user. Browsing through cores may also be
10 permitted, such as with a raster or snake pattern through the tissue microarray. A random mode may also be provided, in which the system randomly presents cores to user.

Fig. 4 is a flow chart of a process for capturing, processing, and storing images of disks in a tissue microarray. It will be appreciated that, while disks are a common geometry
15 used for tissue microarrays, other geometries are possible, including regular and irregular geometric profiles, and may be used with the system described herein, provided they are amenable to punching of matching shapes in a tissue source (for taking samples) and a block of paraffin or similar material (for receiving the samples). The terms ‘disk’ or ‘core’, as used herein, are intended to include any such geometry. The terms ‘specimen’ or ‘biological
20 specimen’ are intended to refer to any biological (or inert control) material that may be sampled and inserted into a tissue microarray.

The process 400 may be realized in hardware, software, or some combination of

these. The process 400 may be realized in one or more microprocessors, microcontrollers, embedded microcontrollers, programmable digital signal processors or other programmable device, along with internal and/or external memory such as read-only memory, programmable read-only memory, electronically erasable programmable read-only memory, random access memory, dynamic random access memory, double data rate random access memory, Rambus direct random access memory, flash memory, or any other volatile or non-volatile memory for storing program instructions, program data, and program output or other intermediate or final results. The process 400 may also, or instead, include an application specific integrated circuit, a programmable gate array, programmable array logic, or any other device that may be configured to process electronic signals.

Any combination of the above circuits and components, whether packaged discretely, as a chip, as a chipset, or as a die, may be suitably adapted to use with the systems described herein. It will further be appreciated that the below process 400 may be realized as computer executable code created using a structured programming language such as C, an object-oriented programming language such as C++ or Java, or any other high-level or low-level programming language that may be compiled or interpreted to run on one of the above devices, as well as heterogeneous combinations of processors, processor architectures, or combinations of different hardware and software. The process 400 may be deployed using software technologies or development environments including a mix of software languages, such as Microsoft IIS, Active Server Pages, Java, C++, Oracle databases, SQL, and so forth.

The process 400 starts 402 with a calibration of the tissue microarray image, as

shown in step 404. A user interface may be provided to assist with the calibration, which may depend on the particular specimen under study and the particular microscope being used. For example, color may be calibrated to accommodate measurement of protein expression for a full spectrum of stains and biologic targets (e.g. stromal, epithelial cells).

5 In this example, the system may perform a mapping of one or more red, green, and blue intensity values of an imaged microarray into L*U*V* color space and then, using polar coordinates, plot the mapped values into an graphical window equipped with interactive controls while a crude multidimensional segmentation of the digitized microarray is performed. Using the graphical controls a user may interactively refine the segmentation by
10 sketching lines of demarcation between clusters within the polar plot while a continuously updated output image shows the effect of utilizing the new parameters. Once the user is satisfied with the segmentation for one disk, the calibration may be applied to the remaining disks on the microarray. These and other known calibration techniques may be used to normalize image data across a number of different tissue microarrays.

15

Once the system is calibrated, the disks in the tissue microarray may be registered, as shown in step 406. The rows and columns of disks in the microarray are rarely straight, and slight distortions to each disk are typically introduced during specimen preparation. To account for this, the system may register each disk to ensure accurate stage localization.
20 Slight errors in lens co-focal and co-centering may be compensated for using empirical data. An entropy-based, fast auto-focusing algorithm may be applied to ensure image quality. The tissue microarray may be, for example, scanned at a magnification of 1.25x. Scanned images may be further scaled down and joined into a map image of the entire tissue microarray.

Given the approximate core diameter (a known parameter for the tissue microarray), the system may automatically generates a disk template, which is approximately the same size as the disks in the mapped image. A template matching protocol is implemented to identify disks based upon one or more visual features of the disks. This matching to visual features may be accomplished by first convolving the map image with the template, using a two-dimensional, discrete convolution, and then performing a top-hat peak-detection for disk centers. The template for the convolution is preferably a circular template based upon the known, approximate core diameter. More specifically, the template is preferably a circle filed with a unit value (e.g., one) for each pixel within the circle, and a circumferential border of negative unit values (e.g., -1). The convolution is preferably accomplished through a multiplication of a two-dimensional, discrete transform of the template with a two-dimensional, discrete transform of the scanned image. The result, after peak detection, is the coordinates of all candidate cores.

As noted above, some deformation of the tissue microarray is expected. To address this issue, rows and columns of candidate cores may be located using a modified Hough transformation algorithm. The Hough transformation may be used to obtain gridlines connecting the cores that have been located into the rows and columns of the (approximately) rectangular array described above. The resulting gridlines may be used to recover positions of the cores that do not include visual features of the matching template. More specifically, grid intersections for which no disk was located using the matched filter above may nonetheless be identified as cores, with an image of each such core captured using the known core diameter of cores in the tissue microarray. This approach may recover, for example, one

or more cores that are not positively stained in the array. By locating these cores with grid intersections, accurate stage coordinates of all cores may thus be recorded for automatic image acquisition or assisted conventional microscopic browsing. Other techniques for locating shapes are known, and may be usefully employed with the systems and methods described herein. However, the above described approach has empirically proven well-suited to use with disks in a tissue microarray. It will be appreciated that modifications will be appropriate for other arrays that are not arranged into a rectangular matrix of samples having regular rows and columns.

Once disks have been located, the process 400 may commence disk image acquisition, as shown in step 408. Using the location data obtained above, the imaging device may be automatically directed to acquire an image of each disk at a higher magnification. The process 400 may auto-focus and background-correct each disk when the image is captured. Auto-focusing may be, for example, through entropy minimization. In order to enhance image detail, the imaging device may capture images of subsections of a disk at higher magnification, which may then be combined to form a single, high-detail image.

After each disk image has been acquired, the images may be analyzed, as shown in step 410. This may be any quantitative or other objective analysis that may be realized in computer software. The images may be processed, for example, into their constituent visual components (e.g. Stromal, epithelial cell regions). The system may then produce measures to determine the signal strength for protein expression (intensity) per unit area and also in

terms of integrated density of protein expression. Additionally, measures for multi-resolution texture and morphometric measurements may be generated, as well as any other useful quantitative measure that may be derived from the images, including measures of shape, size, color, color gradient, contrast, and so forth.

5

As shown in step 412, images and image data, such as image location and the quantitative evaluations discussed above, may be archived. This may be performed automatically, with images and associated data being stored in one or more local and/or distributed relational databases. The commercially available Oracle 8i database system is one database suitable for use with the number and size of records typically encountered in the images contemplated herein. It will be appreciated that each of the steps of disk image acquisition 408, disk analysis 410, and data archiving 412 may be performed in parallel for all disks on a tissue microarray, for groups of disks such as rows, or individually for each disk, and repeated as appropriate until all disks on the tissue microarray are processed. The order in which disks are processed may depend on memory and processing constraints of the system employed, or upon programming convenience. In one embodiment, each disk is processed individually and fed to a database before the next disk in the tissue microarray is analyzed.

20 Once data has been archived in step 412, data may be managed, as shown in step 414. It will be appreciated that this step may be performed immediately upon completion of step 412, or at some subsequent time at a user's convenience. The system may allow a user to design the data format for new tissue microarrays with options for labeling the disks

individually or in groups. The interface may also allow for color coding of the elements (disks) from each subset and for arranging cases. Disk images, and the associated data (such as image metrics and protein expression levels) may also be managed across a number of tissue microarrays and cohorts. Thus new, virtual tissue microarrays may be created from
5 disparate sets of archived data, thereby facilitating the design of new experiments from ensembles of existing cases.

As shown in step 416, the process 400 may end, with a structured database of results available for review by clinicians and/or researchers at local or remote locations.
10

It will be appreciated that the above process is merely illustrative, and that other steps and procedures, or system features, may be usefully deployed with a system as described herein, in addition to, or instead of, those disclosed herein. For example, missing disks may be located through direct inspection of the convolution results, and in certain circumstances,
15 calibration may be omitted.

In one embodiment, the steps of the process 400 are performed by a computer locally connected to a robotic microscope. In another embodiment, the steps of the process 400 are performed by a computer that communicates with the robotic microscope through an internetwork. In either embodiment, access to the image archives may be provided to remote
20 clients through the internetwork. A voice-activated user interface may be provided to simplify computer control over the archiving process, or over review of archived data.

Thus, while the invention has been disclosed in connection with the preferred embodiments shown and described in detail, various modifications and improvements thereon will become readily apparent to those skilled in the art. It should be understood that all matter contained in the above description or shown in the accompanying drawings shall
5 be interpreted as illustrative, and not in a limiting sense, and that the following claims should be interpreted in the broadest sense allowable by law.

What is claimed is: